VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ
BRNO UNIVERSITY OF TECHNOLOGY

FAKULTA ELEKTROTECHNIKY A KOMUNIKAČNÍCH TECHNOLOGIÍ
FACULTY OF ELECTRICAL ENGINEERING AND COMMUNICATION

ÚSTAV TELEKOMUNIKACÍ
DEPARTMENT OF TELECOMMUNICATIONS

VYUŽITÍ STATICKÝCH METOD PRO DETEKCI DDOS ÚTOKŮ
STATIC METHODS FOR DETECTION DDOS ATTACKS

BAKALÁŘSKÁ PRÁCE
BACHELOR'S THESIS

AUTOR PRÁCE — **Lukáš Miško**
AUTHOR

VEDOUCÍ PRÁCE — **Ing. Petr Blažek**
SUPERVISOR

BRNO 2017

VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ
FAKULTA ELEKTROTECHNIKY A KOMUNIKAČNÍCH TECHNOLOGIÍ

Bakalářská práce

bakalářský studijní obor **Teleinformatika**
Ústav telekomunikací

Student: Lukáš Miško ***ID:*** 173708
Ročník: 3 ***Akademický rok:*** 2016/17

NÁZEV TÉMATU:

Využití statických metod pro detekci DDoS útoků

POKYNY PRO VYPRACOVÁNÍ:

Bakalářská práce je zaměřena na využití statických metod, zejména analýzy časových řad, pro identifikaci záplavových DDoS útoků. Úkolem bakalářské práce je prozkoumat vliv záplavových DDoS útoků na kvantitativní parametry datové komunikace (průměrná délka paketu, poměr TCP/UDP atd.). Cílem práce je vytvořit software, který bude schopen detekovat zvolené záplavové DDoS útoky.

DOPORUČENÁ LITERATURA:

[1] PETER J. BROCKWELL, RICHARD A. DAVIS., Peter J. Brockwell, Richard A. Davis. Introduction to time series and forecasting. 2nd ed. New York: Springer, 2002. ISBN 978-038-7216-577.

[2] ENDORF, Carl. Detekce a prevence počítačového útoku. 1. vyd. Praha: Grada, 2005, 355 s. ISBN 80-24--1035-8.

Termín zadání: 1.2.2017 ***Termín odevzdání:*** 8.6.2017

Vedoucí práce: Ing. Petr Blažek
Konzultant:

doc. Ing. Jiří Mišurec, CSc.
předseda oborové rady

ABSTRAKT

Táto práca obsahuje teoretický základ pre riešenie problematiky sieťových anomálii požitím statických metód a taktiež obsahuje riešenie v podobe programu na detekciu sieťových útokov. Zameranie práce je hlavne na detekciu útokov DoS (odmietnutie služby – Denial of Service). V práci sa nachádza rozbor kategorizácie miery vysielania paketov DoS útokov. Ďalej sa v práci nachádza rozbor protokolov TCP (protokol riadenia prenosu – Transmission Control Protocol) a UDP (užívateľský datagramový protokol – User Datagram Protocol), ich možné využitie k útokom SYN záplavy a UDP záplavy. V rámci práce sú rozoberané tri statické metódy a ich detailný popis. V práci sa ďalej nachádza analýza získaných údajov a ich porovnanie. V závery práce sa nachádza popis a výsledky testovania programu vytvoreného k detekcii útokov v sieti.

KĽÚČOVÉ SLOVÁ

DoS, DDoS, DRDoS, SYN, UDP, záplava, detekcia

ABSTRACT

This thesis contains a theoretical basic for solution to issue of network anomalies with use of static methods and it also contains software as a solution for detection of network attacks. The main point of thesis is detection of DoS (Denial of Service) attacks. In thesis is located an analysis of DoS attacks rate categorization. Further in thesis is located analysis of protocols TCP (Transmission Control Protocol) and UDP (User Datagram Protocol), their possible use to attacks SYN flood and UDP flood. Here are analysed three static methods and their detailed description. There is also a analysis of collected data and their comparison in the thesis. Thesis contains description and the results testing of software which is used to detect attacks in network, at the end.

KEYWORDS

DoS, DDoS, DRDoS, SYN, UDP, flood, detection

MIŠKO, Lukáš. *Využití statických metod pro detekci záplavových DDoS útoků*. Brno, 2017, 46 s. Bakalárska práca. Vysoké učení technické v Brně, Fakulta elektrotechniky a komunikačních technologií, Ústav telekomunikací. Vedúci práce: Ing. Petr Blažek,

VYHLÁSENIE

Vyhlasujem, že som svoju bakalársku prácu na tému „Využití statických metod pro detekci záplavových DDoS útoků“ vypracoval(a) samostatne pod vedením vedúceho bakalárskej práce, využitím odbornej literatúry a ďalších informačných zdrojov, ktoré sú všetky citované v práci a uvedené v zozname literatúry na konci práce.

Ako autor(ka) uvedenej bakalárskej práce ďalej vyhlasujem, že v súvislosti s vytvorením tejto bakalárskej práce som neporušil(a) autorské práva tretích osôb, najmä som nezasiahol(-la) nedovoleným spôsobom do cudzích autorských práv osobnostných a/alebo majetkových a som si plne vedomý(-á) následkov porušenia ustanovenia § 11 a nasledujúcich autorského zákona Českej republiky č. 121/2000 Sb., o práve autorskom, o právach súvisiacich s právom autorským a o zmene niektorých zákonov (autorský zákon), v znení neskorších predpisov, vrátane možných trestnoprávnych dôsledkov vyplývajúcich z ustanovenia časti druhej, hlavy VI. diel 4 Trestného zákoníka Českej republiky č. 40/2009 Sb.

Brno ....

podpis autora(-ky)

POĎAKOVANIE

Rád bych poděkoval vedoucímu diplomové práce panu Ing. Petrovi Blažekovi, za odborné vedení, konzultace, trpělivost a podnětné návrhy k práci.

Brno ....

podpis autora(-ky)

Faculty of Electrical Engineering
and Communication
Brno University of Technology
Purkynova 118, CZ-61200 Brno
Czech Republic
http://www.six.feec.vutbr.cz

POĎAKOVANIE

Výzkum popsaný v tejto bakalárskej práci bol realizovaný v laboratóriách podporených projektom SIX; registračné číslo CZ.1.05/2.1.00/03.0072, operačný program Výzkum a vývoj pro inovace.

Brno ....

podpis autora(-ky)

EVROPSKÁ UNIE
EVROPSKÝ FOND PRO REGIONÁLNÍ ROZVOJ
INVESTICE DO VAŠÍ BUDOUCNOSTI

OBSAH

Úvod **11**

1 Útoky DoS, DDoS a DRDoS **12**
1.1 Denial of Service 12
1.2 Distributed Denial of Service 12
1.3 Distributed Reflector Denial of Service 14

2 Najčastejšie typy záplavových útokov **15**
2.1 Protokol UDP 15
2.1.1 UPD flood 16
2.2 Potokol TCP 16
2.2.1 Komunikačný model TCP 16
2.2.2 SYN flood 18

3 Kategorizácia Distributed Denial of Service útoku podľa miery výskytu paketov **20**
3.1 Útok nízkej miery výskytu paketov 20
3.2 Útok narastajúcej miery výskytu paketov 21
3.3 Útok konštantnej miery výskytu paketov 21

4 Statické metódy **23**
4.1 Detekcia sledovania zdrojovej IP adresy 23
4.2 Detekcia štatistickým testom 24
4.2.1 Štatistická analýza a detekčné schéma 25
4.3 Štatistická segregačná metóda 26
4.3.1 Vzorkovacia metóda 26
4.3.2 Analýza miery výskytu paketov 27
4.3.3 Korelačná analýza 28

5 Analýza zachytených útokov SYN a UDP záplavy **29**
5.1 Analýza hodinového toku 29

6 Návrh a popis programu **34**
6.1 Učiaca časť 34
6.2 Detekčná časť 37

7 Testovanie programu v linuxovej distribúcii Kali **40**

8 Závěr **43**

Literatúra **44**

Zoznam symbolov, veličín a skratiek **46**

ZOZNAM OBRÁZKOV

1.1 Štruktúra typického útoku Distributed Denial of Service (DDoS) . . . 13
1.2 Štruktúra typického útoku Distributed Denial of Service (DRDoS)[2] 14
2.1 Three-way handshake 17
2.2 Ignorovanie SYN-ACK žiadostí útočníkom 18
3.1 Graf útoku nízkej miery výskytu paketov 20
3.2 Graf útoku narastajúcej miery výskytu paketov 21
3.3 Graf útok konštantnej miery výskytu paketov 22
4.1 Hash table určený pre detekčnú časť 24
4.2 Diagram identifikovania prevádzky 26
5.1 Umiestnenie záznamových sond v infraštruktúre Cesnetu 29
5.2 Graf celkového počtu paketov prevádzky od 19:00 do 20:00 hodiny . . 30
5.3 Graf počtu TCP paketov s príznakom SYN od 19:00 do 20:00 hodiny 31
5.4 Graf počtu UDP paketov od 19:00 do 20:00 hodiny 31
5.5 Graf testovacieho útoku SYN záplavy 32
5.6 Graf testovacieho útoku UDP záplavy 33
6.1 Výpis z terminálu pri úspešnom vytvorení štatistiky 35
6.2 Vytvorený súbor štatistík z pcap v cykle 30 sekúnd 36
6.3 Detekovanie toku dát na porte eth0 37
6.4 Detekcia 20% vzrastu ACK,FIN A SYN na porte eth0 39
6.5 Detekcia vzrastu hodnôt o viac ako 50% na porte eth0 39
7.1 Zapojenie virtualnych strojov na PC pomocou VMware 40
7.2 Spustený program bez známok anomálii 41
7.3 Zaznamenanie anomálie v rozmedzí 20% až 50% 41
7.4 Hodnoty zapísané z terminálu do logu 41
7.5 Zaznamenanie útoku pred a po prekročení 50% 42

ZOZNAM TABULIEK

2.1 Hlavička UDP 15
2.2 Príznaky v TCP protokole 17

ÚVOD

V modernej dobe internetu sa čoraz viac stretávame s anomáliami v dátovej prevádzke, ktoré majú viacero príčin. Môže sa jednať o nárast dátovej komunikácie v dôsledku významných udalostí akými sú napríklad olympijské hry, majstrovstvá v hokeji alebo futbale a podobne. Takýto nárast komunikácie býva časovo obmedzený a dátová prevádzka sa po ukončení udalosti vráti do svojej pôvodnej podoby.

Najčastejšími príčinami anomálii, s ktorými sa môžeme v dátovej prevádzke stretnúť, sú útoky typu DoS (odmietnutie služby – Denial of Service), DDoS (distribuované odmietnutie služby – Distributed Denial of Service) a DRDoS (distribuovano reflektované odmietnutie služby – Distributed Reflector Denial of Service). Ich účelom je znemožnenie komunikácie určitého uzlu v sieti alebo zahltenie siete a tým odoprenie služby ostatným používateľom, keďže nie je možnosť spracovávať požiadavky týchto používateľov. Dôsledky týchto útokov sa môžu líšiť od nepríjemností pre užívateľov internetových stránok až po značné finančné straty pre firmy, ktoré sa spoliehajú na dostupnosť ich on-line služieb.

Detekcia týchto anomálii bude v rámci tejto bakalárskej práce skúmaná pomocou statických metód a analýzou časových rád. V časti teoretického riešenia budú rozoberané protokoly TCP (protokol riadenia prenosu – Transmission Control Protocol), UDP (užívateľský datagramový protokol – User Datagram Protocol), ktoré využívajú útoky DoS, DDoS a DRDoS k účelu znemožnenia komunikácie.

V praktickej časti bude popísaný vytvorený program na detekciu útočného toku, budú uskutočnené testy vďaka ktorým sa bude dokazovať funkčnosť programu a jeho schopnosti detekcie útokov.

1 ÚTOKY DoS, DDoS, DRDoS

Útoky DoS, DDoS a DRDoS, ktoré sú popísané v tejto kapitole, sú útokmi odoprenia služby, anglicky Denial of Service.

1.1 Denial of Service

DoS (odmietnutie služby – Denial of Service) je útok, ktorý pochádza od jedného útočníka. Tento typ útoku je veľmi populárny pre jeho jednoduchosť a efektivitu. Je to typ útoku na internetové služby alebo stránky, ktorého cieľom je vyčerpať všetky zdroje cieľa útoku. Môže ísť o vyčerpanie prenosového pásma, výpočtového výkonu alebo obmedzenia dátových štruktúr operačného systému. Hlavným zámerom útočníka je narušiť kritické služby cieľa, ktoré sú napríklad:

- elektronické bankovníctvo,
- email,
- prístup k webovým stránkam.

Týmto spôsobom sa útočník snaží znemožniť platným užívateľom týchto služieb bezproblémový prístup. V niektorých prípadoch DoS útoky prinútili web stránky na ktorých boli pripojených milióny účastníkov k dočasnému odstaveniu prevádzky[1].

1.2 Distributed Denial of Service

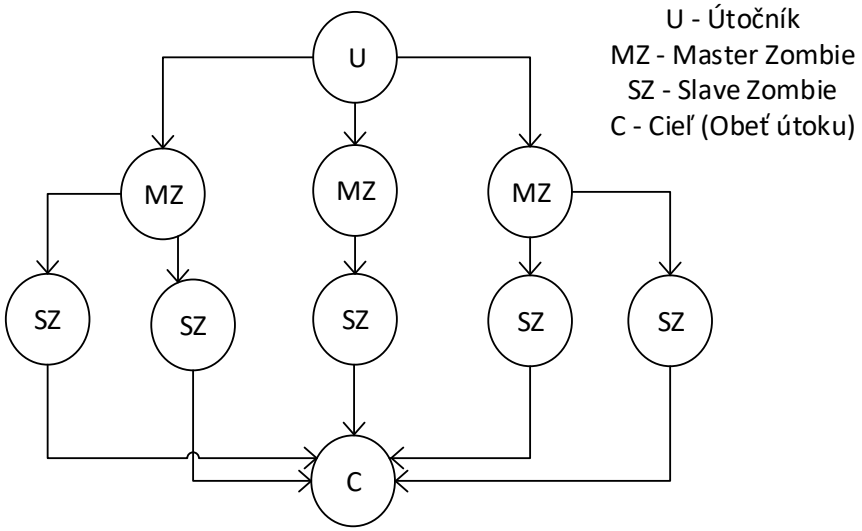
DDoS (distribuované odmietnutie služby – Distributed Denial of Service) tento útok je jednou z hlavných hrozieb internetu pre jeho masívny koordinovaný útok, ktorý využíva veľa rôznych zdrojov odkiaľ sú zasielané neužitočné pakety k cieľu vo veľmi krátkom čase, čo spôsobí vyčerpanie zdrojov cieľa a cieľovú službu spraví neprístupnou. Pomocou DDoS útokov je jednoduchšie vykonávať škodlivé, náročne vystopovateľné a ťažko zabrániteľné útoky.

Ak chce útočník spustiť DDoS útok musí najprv vybudovať sieť počítačov, ktoré využije k vyprodukovaniu veľkého množstva prevádzky voči serveru na ktorom beží daná služba, ktorú chce útočník vyradiť z prevádzky. K vytvoreniu tejto útočnej siete počítačov útočník vyhľadáva zraniteľné počítače v sieti. Zraniteľné počítače v sieti sú väčšinou tie, ktoré nemajú žiadny antivírusový software, antivírusový software nie je aktualizovaný alebo tieto počítače nemajú spravené aktualizácie systému a tak nie sú zabezpečené systémové chyby, ktoré už boli odhalené. Zraniteľný hostitelia sú potom napadnutý a využívaný útočníkom, ktorý prevezme kontrolu nad ich zariadeniami.

Útočníkov ďalší krok je inštalácia nových programov, útočných programov, do napadnutých zariadení. Z napadnutých hostiteľov sa takto stávajú *zombie*, ktorý sú

pod útočníkovou kontrolou. Keď útočník napadne veľké množstvo počítačov, z ktorých sa stanú zombie vybuduje si tak armádu útočníkov, ktorých využíva k uskutočneniu DDoS útoku na daný server alebo službu na servery.

V typickom DDoS útoku sa armáda útočníka skladá z *„master zombie“* a *„slave zombie“*. V oboch prípadoch sa jedná o hostiteľov, ktorý sú napadnutý škodlivým kódom. Útočník koordinuje a udeľuje príkazy pre master zombies a oni ďalej koordinujú a spúšťajú slave zombies. Presnejšie povedané útočník odošle príkaz na útok pre master zombies, tí aktivujú všetky procesy a služby, ktoré sú v režime hibernácie a čakajú na správny príkaz od útočníka. Master zombies pomocou týchto procesov pošlú príkaz na útok pre slave zombies, v ktorom im prikážu zasielať DDoS útok namierený na obeť. Takýmto spôsobom začnú slave zombies posielať veľké množstvo paketov voči obeti útoku, zahltia jej systém zbytočnou záťažou a vyčerpajú zdroje obete.



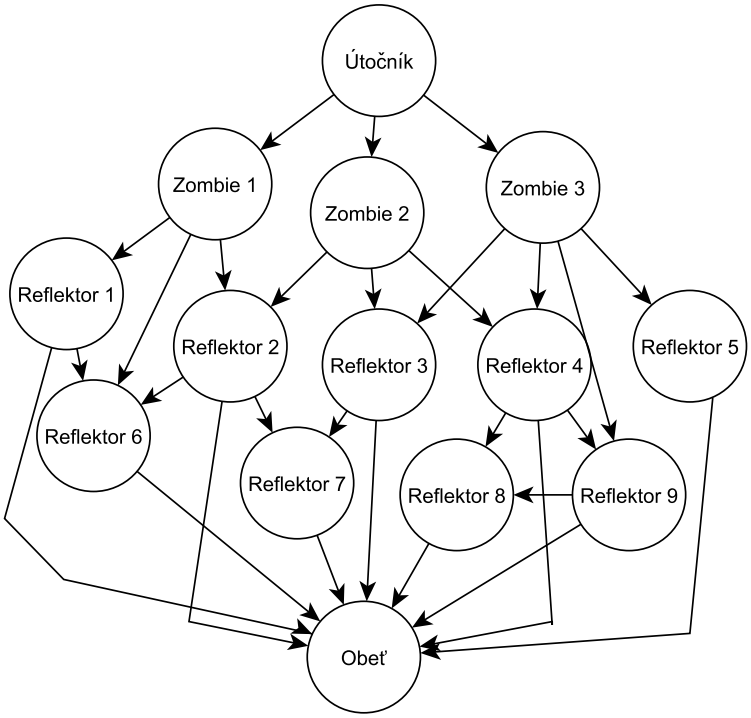
Obr. 1.1: Štruktúra typického útoku Distributed Denial of Service (DDoS)

1.3 Distributed Reflector Denial of Service

DRDoS (distribuovano reflektované odmietnutie služby – Distributed Reflector Denial of Service) je ďalším typom DDoS útoku zobrazeným na obrázku 1.2, ktorý využíva zariadenia tretích strán (*routre alebo web servery*) na odrážanie útočnej prevádzky proti obeti útoku. DRDoS útok pozostáva z troch štádií:

- Prvé štádium je rovnaké ako aj pre typický **DDoS** útok kde útočník infikuje škodlivým kódom obete z ktorým sa následne stávajú zombies.
- V druhom štádiu namiesto toho aby útočník nariadil zombies útok na obeť, ako pri normálnom **DDoS** útoku, útočník nariadi zombie hostiteľom rozosielať falošné požiadavky na vytvorenie komunikácie so zdrojovou IP adresou obete na zariadenia tretích strán.
- V treťom štádiu zariadenia tretích strán začnú odosielať odpovede na požiadavky o vytvorenie komunikácie od zombies so zdrojovou IP adresou obete, pretože si myslia, že komunikácia bola vyžiadaná a takto zahltia obeť nevyžiadanou prevádzkou.

V tomto štádiu sa jedná už o klasický **DDoS** útok. Tento typ útoku zapríčinil výpadok *www.grc.com* v januári 2002, stránky zaoberajúcej sa výskumom bezpečnosti a je považovaný za účinný, stále rozšírenejší a znepokojujúci internetový útok[2].



Obr. 1.2: Štruktúra typického útoku Distributed Denial of Service (DRDoS)[2]

2 NAJČASTEJŠIE TYPY ZÁPLAVOVÝCH ÚTOKOV

Záplavové útoky spoznáme väčšinou podľa toho, že v ich mene majú slovo záplava (*flood*).[3] Tieto útoky sú medzi sebou veľmi podobné, líšia sa hlavne použitým protokolom. Najčastejšie dva typy záplavových útokov :

- UDP flood (*UDP záplava*),
- SYN flood (*SYN záplava*),

2.1 Protokol UDP

UDP (užívateľský datagramový protokol – User Datagram Protocol) je komunikačný protokol používaný napríklad pre aplikácie ako DNS, DHCP, TFTP, VoIP. Model UDP protokolu je omnoho jednoduchší než TCP. Protokol nerobí žiadnu kontrolu chyb relačnej vrstvy a nemá ani zabudovaný mechanizmus znovu odosielanie stratených paketov, ale pakety obsahujú kontrólny súčet. UDP využíva 16bitové čísla portov tak ako TCP, ale UDP a TCP porty sú úplne odlišné.[4]

Hlavička UDP sa skladá zo štyroch 16bitových polí o celkovej dĺžke 8bajtov:

Zdrojový port	**Cieľový port**
Dĺžka	**Kontrolný súčet UDP**

Tab. 2.1: Hlavička UDP

- **Zdrojový port** – je 16 bitové číslo (0 až 65 535), port aplikácie, ktorá vytvára datagram.
- **Cieľový port** – je 16 bitové číslo (0 až 65 535), port aplikácie, ktorá má prijať datagram.
- **Dĺžka** – je 16 bitové číslo (0 až 65 535), pole je dĺžkou hlavičky UDP a dátových bitov, ktoré nezahrnuje hlavičku IP.
- **Kontrólný súčet** – je 16 bitové číslo (0 až 65 535), nie je povinný, ak je pole nastavené na 0 prijímací uzol kontrolný súčet nekontroluje.

2.1.1 UPD flood

Hlavička UDP je tak primitívna, že sa javí ako naproste odolná proti zneužitiu. Obsahuje iba zdrojový a cieľový port, dĺžku paketu a kontrolný súčet 2.1. Kontrolný súčet UDP sa vypočítava len voliteľne, ak je toto 16bitové pole rovné 0, znamená to, že kontrolný súčet pre daný prenos nebol vypočítaný a tak by sa príjemca nemal kontrolovať.

UDP flood je typ DoS útoku pomocou ktorého sa útočník snaží preťažiť náhodné porty na cieľovom počítači s IP paketmi, ktoré obsahujú UDP datagramy. Cieľ útoku tieto pakety kontroluje a snaží sa im priradiť aplikácie spojené s týmito datagramy avšak odosiela naspať príjemcovi paket „*Destination Unreachable* “ – cieľ nedosiahnuteľný. Ako ďalej cieľ prijíma a odpovedá na ďalšia a ďalšie UDP pakety stáva sa preťažený a nedostupný ostatným používateľom.

Útočník môže pri útoku falšovať svoju pravú IP adresu a tak zaistiť, že žiadny z vyslaných paketov sa nevráti naspať k útočníkovi. Existuje veľké množstvo zdarma stiahnuteľných nástrojov, ktoré môžu byť použité k vykonaniu UDP flood útoku ako napríklad *UDP Unicor* alebo *Low Orbit Ion Cannon*[5].

2.2 Potokol TCP

TCP (protokol riadenia prenosu – Transmission Control Protocol) je protokol, ktorý je klenotom z rodiny IP protokolov a jeho hlavnou úlohou je spoľahlivý prenos dát.[4] Transmission Control Protocol je spoľahlivým protokolom, ktorý narozdiel od UDP protokolu vypočítava kontrolný súčet a obsahuje tri vzájomne odlišné spolupracujúce mechanizmy:

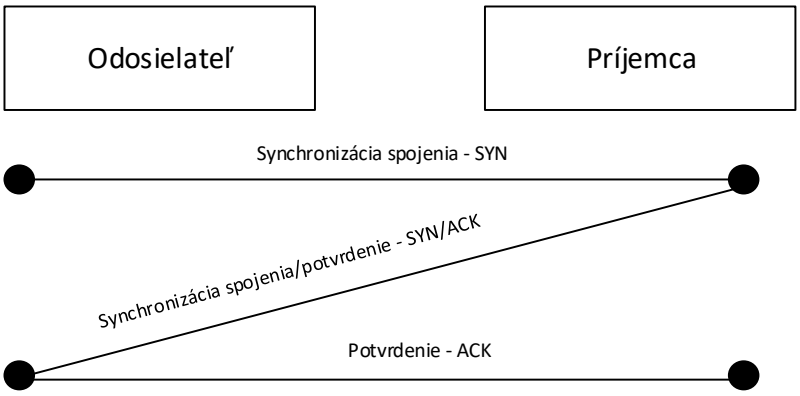
- **Kontrolný súčet** Kontrolný súčet je vypočítavaný cez celý TCP paket vrátane dátovej časti.
- **Vzájomné odsúhlasovanie prijatých dát** Každý systém si udržiava prehľad o počte správne prijatých bajtov od vysielacej strany (ACK súčet), tak ako aj o počte vysielaných bajtov smerom k prijímacej strane (SYN počet).
- **Vypršanie doby platnosti paketu** Keďže IP negarantuje dodávku, môže nastať prípad keď sa vypustí paket bez upozornenia. TCP určitú dobu čaká na odpoveď na odoslaný paket a potom ho odošle znova, pretože je v domnení, že sa paket stratil alebo sa stratilo potvrdenie.

2.2.1 Komunikačný model TCP

Každé spojenie TCP je jedinečne identifikované pomocou štyroch záznamov: IP adresou dvoch komunikujúcich systémov a dvoma číslami TCP portov používaných každou stranou.

Za normálnych okolností musí byť na strane príjemcu aktívny odposlúchajúci proces aby bol schopný prijať a odpovedať na TCP požiadavku o spojení. Pre nadviazanie spojenia TCP využíva takzvaný *„three-way handshake“*, ktorý je popísaný v nasledujúcich krokoch zobrazených na obrázku 2.1:

1. Príjemca je v pasívnom odpočúvacom stave a prebudí sa až po prijatí požiadavku od odosielateľa na synchronizáciu spojenia.
2. Odosielateľ vysiela začiatočný TCP paket, ktorý má nastavený príznak SYN a svoje počiatočné sekvenčné číslo(ISN, Initial Sequence Number), ktoré je generované pseudonáhodne. Každý prenesený bit, vrátane príznakov (SYN alebo FIN), toto sekvenčné číslo inkrementuje[4].
3. Príjemca na tento paket odpovedá SYN/ACK paketom, ktorým oznamuje odosielateľovi svoju pripravenosť k nadviazaniu spojenia.
4. Po zaslaní paketu s príznakom ACK od odosielateľa k príjemcovi sa ukončí fáza, ktorá sa nazýva *„three-way handshake“*.



Obr. 2.1: Three-way handshake

Pri komunikácii sa v protokole využívajú príznaky popísané v tabuľke 2.2.

Názov príznaku	**Popis**
URG	Príznak vyjadruje naliehavosť paketu
ACK	Príznak potvrdenia
PHS	Príznak posunutia
RST	Príznak resetovania spojenia
SYN	Príznak synchronizácie spojenia
FIN	Príznak vyjadrujúci koniec spojenia

Tab. 2.2: Príznaky v TCP protokole

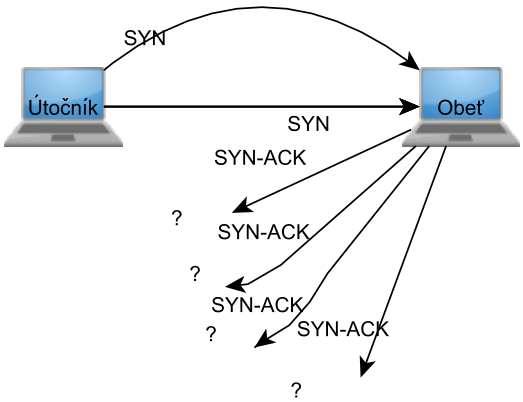
2.2.2 SYN flood

SYN flood je typ útoku, ktorý za pomoci TCP protokolu využíva chybovosť v implementácii *„three-way handshake“* k vyčerpaniu zdrojov cieľa (obete útoku), v ktorom útočník zasiela veľké množstvo SYN žiadostí (žiadosti o synchronizáciu spojenia), čo v konečnom dôsledku spôsobuje zahltenia a rozvrat vo využívanej pamäti.

Keď hostiteľský počítač obdrží žiadosť SYN od iného užívateľa odpovedá na túto žiadosť paketom SYN/ACK, tým vytvorí vstup vo fronte neobslúžených žiadostí a vyčkáva na odpoveď, ACK paket, ktorým sa dokončí *„three-way handshake“* popísaný v časti 2.2.1. Poznatok o fungovaní *„three-way handshake“* útočník využíva k zavedeniu útoku.

Útočník zasiela veľké množstvo SYN paketov na otvorený port obete, ktoré sa následne ukladajú do fronty neobslúžených žiadostí. Obeť následne posiela späť pakety SYN/ACK, vytvára ďalší vstup vo fronte neobslúžených žiadostí o spojenie a čaká na dokončenie *„three-way handshake“*. V tejto fázy sa spojenie nazýva *„half open“* (polootvorené) a obeť vo fronte „načúva“ polootvorenému spojeniu. Typický *„timeout“* (časové oneskorenie) pre takéto pripojenie je 3 minúty. Toto umožňuje úspešnému spojeniu aj pri dlhých meškaniach na sieti. Kým toto časové oneskorenie vyprší útočník môže posielať množstvo požiadaviek na pripojenie s falošnými zdrojovými adresami na počítač obete [6][4].

To spôsobí, že zdroje obete útoku sú alokované na spracovanie týchto žiadosti. SYN/ACK pakety obete útoku sú útočníkom ignorované, ako je možné vidieť na obrázku2.2. Keď sa fronta naplní, systém nie je schopný prijať žiadny ďalší požiadavok o spojenie aj v prípade, že tento požiadavok je od obyčajného užívateľa.



Obr. 2.2: Ignorovanie SYN-ACK žiadostí útočníkom

Ciele útoku môžu využiť stratégie proti útoku a to buď jednotlivo alebo aj kombináciou viacerých stratégii na vyrovnanie sa s útokmi odoprenia služby.

- **Obmedzenie rýchlosti spojenia** Ciele útokov môžu zaviesť software, ktorý obmedzuje rýchlosť spojenia v danom časovom rámci. Uzol na ktorý sú posielané žiadosti potom jednoducho odmieta akékoľvek žiadosti presahujúce jeho limit. Nevýhodou spomínanej stratégie je možné odmietanie legitímnych žiadostí o nadviazanie spojenia.
- **Ošetrenie expirovaného spojenia** Keď sa tabuľka spojení zaplní práve prijatým požiadavkom, niektoré uzly z obslužnej fronty odstraňujú náhodne vybrané expirované požiadavky. Táto stratégia vkusne rieši problémy s útokmi, pretože staré požiadavky môžu pravdepodobne pochádzať skôr zo SYN záplavy ako od legitímnych žiadostí o spojenie. Avšak možne nastať situácie kde aj legitímne spojenie môže byť odstránené zo žiadostí o spojenie, pretože jeho čas odozvy je krátky alebo rýchlosť na požiadavky extrémne vysoká.
- **Využitie SYN cookies (SYN sušienky)** SYN cookies alebo tiež synchronizačná sušienka je technika v ktorej sa pridáva špeciálna hodnota, pri nadväzovaní TCP komunikácie, do paketu. V nej sa počiatočné sekvenčné číslo ISN (Initial Sequence Number) kryptograficky odvodzuje z IP adries a čísel portov tohto spojenia a naspäť sa zasiela paket SYN/ACK s týmto ISN. Toto zaisťuje možnosť pre server na komunikáciu zabudnúť až do chvíle, kým neobdrží ACK paket s rovnakým príznakom ISN. Server nemusí držať spojenie polootvorené a tak zbytočne vyčerpávať svoje zdroje. Keď server prijme ACK paket, ktorý má v sebe implementovaný SYN cookies a zistí, že nie je časťou stanovovaného spojenie, server vykoná rovnaký kryptografický výpočet a výsledná hodnota je podobná s prijatým sekvenčným číslom. Ak sa tieto dve hodnoty vzájomne zhodujú je spojenie považované za ustanovené a pokračuje bežná komunikácia. Najväčšou nevýhodou tejto stratégie je požiadavok na server aby kryptograficky vypočítaval súčet pre každý nevyžiadaný a prijatý ACK paket, čo môže vytvoriť problémy s využitím CPU (centrálna procesorová jednotka – Central Processing Unit) na servery. Tato metóda existuje už od roku 1996, skonštruovanie a formulácie metódy sa pripisuje americkému expertovi a kryptológovi Danielovi J. Bernsteinovi.[7]

3 KATEGORIZÁCIA DISTRIBUTED DENIAL OF SERVICE ÚTOKU PODĽA MIERY VÝSKYTU PAKETOV

Najnovší DDoS boti, sídliaci v napadnutých počítačoch z ktorých sa stávajú zombie, môžu generovať útoky nízkej miery, s narastajúcou mierou, konštantnej miery a s kolísajúcou mierou paketov v záplave.

3.1 Útok nízkej miery výskytu paketov

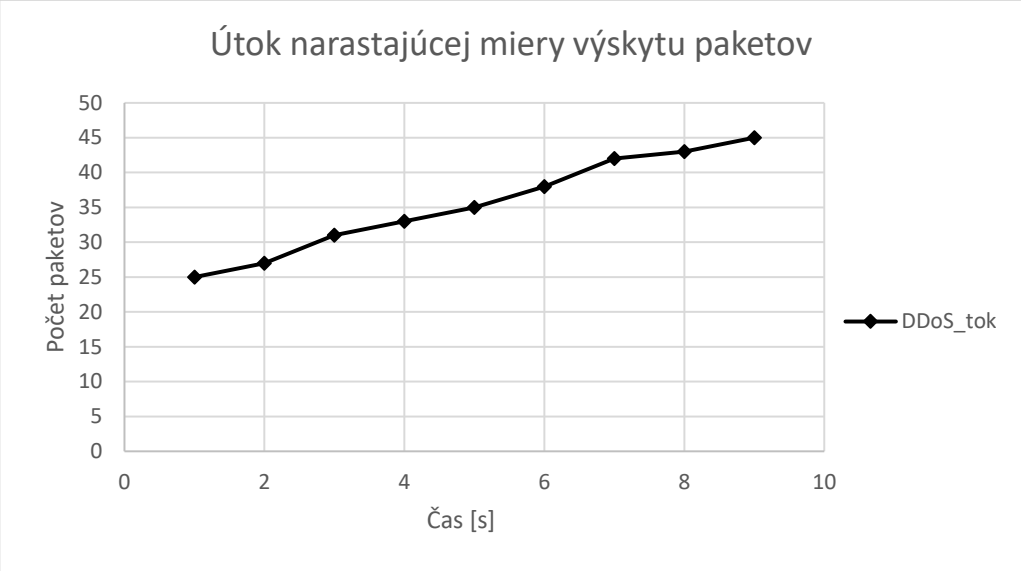
Tento typ mechanizmu maskuje záplavu ako normálnu (legitímnu) prevádzku v sieti, vzhľadom k tomu, že pakety sú generované tak aby ich správanie napodobňovalo skutočného klienta a tak sa zabránilo detekcii, na obrázku 3.1 môžme vidieť ako sa odráža miera prijatých paketov v čase pri tomto type útoku. Prevádzka nikdy nezaplaví šírku pásma v sieti, ale vzhľadom k tomu, že ide o koordinovaný útok tisícok počítačov, ktoré generujú prevádzku voči obeti, môžu preťažiť obeť útoku. Tento útok sa maskuje ako normálny tok, čo sťažuje detekciu detekčným a segregačným mechanizmom.

Obr. 3.1: Graf útoku nízkej miery výskytu paketov

Na druhej strane k vykonaniu takéhoto útoku je za potreby veľké množstvo zombie počítačov a tak sa tento mechanizmus útoku stáva neefektívny z hľadiska nákladov.

3.2 Útok narastajúcej miery výskytu paketov

Miera paketov v tomto type záplavy je neustále narastajúca začínajúc s najmenším možným množstvom paketov, ktoré sa postupne zvyšuje. Vďaka tomuto mechanizmu sa oddaľuje možnosť včasnej detekcie útoku. Zameraním útoku je ochromenie cieľového serveru kúsok pomalšie, ako za pomoci útoku konštantnej miery výskytu paketov, tým si tento útok dáva takpovediac na čas. Na obrázku 3.2 je zobrazený možný útok pomocou narastajúcej miery paketov.



Obr. 3.2: Graf útoku narastajúcej miery výskytu paketov

3.3 Útok konštantnej miery výskytu paketov

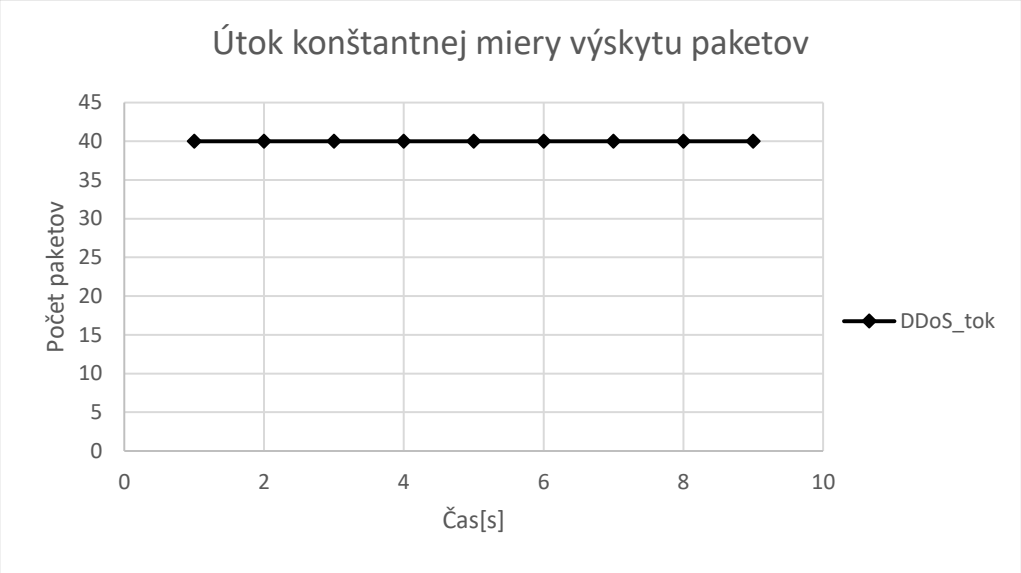
Väčšina skorších útokov bola predvádzaná pomocou mechanizmu vysielania konštantnej miery paketov. Útočník poslal príkaz na zombie počítače, ktorý obsahoval povel na generovanie rovnakého počtu paketov v každom intervale. Tie potom generovali konštantnú mieru prevádzky 3.3, proti obeti útoku v miere väčšej ako bola miera pre legitímnu prevádzku.

Vďaka tejto zvýšenej miere paketov sa zrazu vytvorí záplava paketov, ktorá naruší server obete veľmi rýchlo. Tento spôsob je najlepším pre útočníka z hľadiska nákladov na útok, pretože môže nasadiť minimálne počtov zombie počítačov k spôsobeniu vážnejšieho poškodenia obete útoku.

Avšak miera paketov môže v polovici cesty kúsok klesnúť kvôli stratám v prevádzke, pretože tento útok je polovičným útokom na šírku pásma. Tento útok nie je

priamo zameraný na šírku pásma, no v rovnakom čase útočníkovi nevadí ak aj náhodou preťaží šírku pásma. Tento typ útoku však môže byť segregovaný od normálnej prevádzky, pretože miera paketov, ktoré sú generované je nad normálny limit.

Útoky na báze šírky pásma spôsobujú zahltenie celej siete. Preto sa používajú len v ojedinelých prípadoch, pretože sú všadeprítomné filtre paketov, ktoré môžu detekovať a prípadne aj vyradiť záplavu v samotnej sieti pred dosiahnutím cieľa .



Obr. 3.3: Graf útok konštantnej miery výskytu paketov

V dôsledku moderných útočníkov a útokov sa tomuto typu útoku nedáva prednosť. No aj naďalej sa ešte môžme stretnúť s týmto útokom v praxi, hlavne pri nových útočníkoch, ktorý nevenujú pozornosť pri miere výskytu paketov v útoku [9].

4 STATICKÉ METÓDY

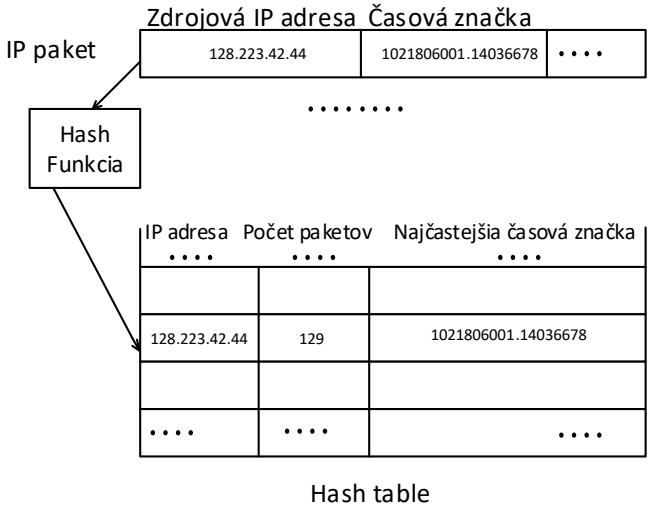
V tejto kapitole budú popísané statické metódy, vďaka ktorým je možno identifikovať DoS (odmietnutie služby – Denial of Service) útoky.

4.1 Detekcia sledovania zdrojovej IP adresy

Táto statická metóda je založená na schéme nazývanej SIM (monitorovanie zdrojovej IP adresy – Source IP address Monitoring) k detekcii veľkého množstva DDoS útoku. V detekčnej schéme sa využíva vlastnosti útoku hlavne veľkého množstva nových IP adries v útočnej prevádzke smerom k cieľu útoku. Výhodou metódy by mala byť jej detekcia bližšie k útočníkovi a vyhodnoteniu útoku v skorom štádiu. Model SIM detekcie pozostáva z dvoch častí:

1. *Off-line tréning* – v časti off-line tréningu sa pridávajú do IAD (databáza IP adries – IP Address Database) autorizované IP adresy a databáza IAD sa stále obnovuje pridávaním nových IP adries a mazaním starých IP adries po expirácii. Celý tento proces je vytváraný off-line aby bolo zaistené, že žiadne dáta obsiahnuté počas tréningu nebudú obsahovať akýkoľvek útok na šírku pásma.
2. *Detekcia a učenie* – v tejto časti sa počas určitej periódy zbiera niekoľko štatistík z prichádzajúcej prevádzky za aktuálny časový interval. V detekčnej časti sa používa hash table 4.1 na zaznamenanie IP adries, ktoré sa objavili v určitom časovom intervale. Každý hash table pozostáva z dvoch polí, počtu IP paketov a časovej značky najčastejšieho paketu pre danú IP adresu.

Porovnaním súčtu v hash table s IAD sa môže vypočítať koľko nových IP adries sa objavilo v tomto časovom intervale. Ak je počet paketov pre IP adresu väčší ako je nastavený aktuálny horný prah počtu paketov, alarm je nastavený na indikovanie útoku na šírku pásma. Tento spôsob sa využíva pri detekcie niektorých nedokonalých útokov, ktoré používajú malý počet zdrojových adries. Avšak hlavne analýza počtu nových IP adries môže detekovať, či dochádza k rozsiahlemu DDoS útoku.[2]



Obr. 4.1: Hash table určený pre detekčnú časť

4.2 Detekcia štatistickým testom

Bolo navrhnutých veľa metód na obranu serverov proti SYN flood útokom. Tieto metódy sa dajú kategorizovať do troch hlavných skupín prístupov:

1. Značenie paketov – prístup značenia paketov je založený na označovaní podozrivých paketov s niektorými bitmi v distribuovaných smerovačoch a na následnej filtrácii paketov, v prípade porušenia nastavených pravidiel alebo prekročenia prahovej hodnoty.
2. Proaktívne – proaktívny prístup väčšinou modifikuje existujúci protokol k zabráneniu diania DDoS útok alebo odlišuje nebezpečnú prevádzku od normálnej za použitia PacketScore, skúškou pravdepodobnosti alebo so zavedením inteligentného rámca.
3. Reaktívne – na druhej strane reaktívny prístup preberá určitú zodpovednosť potom ako je nebezpečná prevádzka detekovaná. Osobitný dôraz sa kladie projektovaniu prahového algoritmu a ako sú ovplyvňované parametre algoritmu. Kombináciou koordinovanej detekcie a reakcie rámca sa tu ukazuje lepší spôsob ako zmierniť dopady spôsobené DDoS.

Veľa výskumov je zameraných na navrhovanie účinných protiopatrení pre detekciu záplavových útokov. Avšak s nízkou mierov výskytu paketov v útoku sa útok môže javiť ako bežný prístup užívateľa, môže klesnúť pod podmienky detekcie, tak oklamať stratégiu na detekciu útoku a zaplňovať frontu nevybavených žiadostí. Narozdiel od ostatných metód obrany sa v detekčnej metóde štatistickým testom navrhuje využívať dva štatistické testy na identifikáciu nebezpečnej prevádzky.

V prvom rade sa porovnávajú celkové hodnoty miery príchodu prevádzky a miery príchodu v normálnej prevádzke na základe dvoch testovacích vzoriek. V prípade že je rozdiel týchto vzoriek významný je veľmi pravdepodobné, že sú v prevádzke zahrnuté útočné záplavové pakety. Avšak útok s nízkou mierov výskytu paketov v toku môže prejsť vstupným testom a zahlcovať frontu nevybavených žiadosti. Preto sa potom porovnávajú dve skupiny, ktoré obsahujú rôzny počet SYN a ACK paketov dvomi **t**–testovacími vzorkami. Ak aj v tomto prípade je rozdiel značný, môžme uznať, že v aktuálnej prevádzke je zamiešaná aj útočná prevádzka. Môžu nastať prípady, kedy je normálna prevádzka vyhodnotená ako útočná *(falošne pozitívna)*, ale aj prípady kde je útočná prevádzka vyhodnotená ako bežná prevádzka *(falošne negatívna)*. Priebeh celej analýzy je zobrazený v diagrame na obrázku4.2.

4.2.1 Štatistická analýza a detekčné schéma

V štatistickej analýze sa najskôr potvrdí SAR (počet prijatých SYN – SYN arrival rates) vzorka distribuovaná z normálnej prevádzky. Následne sa opakovane meria SAR, vypočítava sa stredná aritmetická hodnota ($\overline{X}$) a smerodajná odchýlka ($\widehat{S}$) zo vzorku SAR. Nech X_1, X_2, ..., X_N sú vzorky N meraní. Dostávame

$$\overline{X} = \frac{\sum_{i=1}^{N} X_i}{N}. \tag{4.1}$$

Vzorka rozptylu ($\widehat{S}^2$) pre vzorku N meraní je rovná súčtu štvorcu vzdialeností od stredu delenej $(N-1)$. Z toho vychádza

$$\widehat{S}^2 = \frac{\sum_{i=1}^{N} (X_i - \overline{X})^2}{N-1}. \tag{4.2}$$

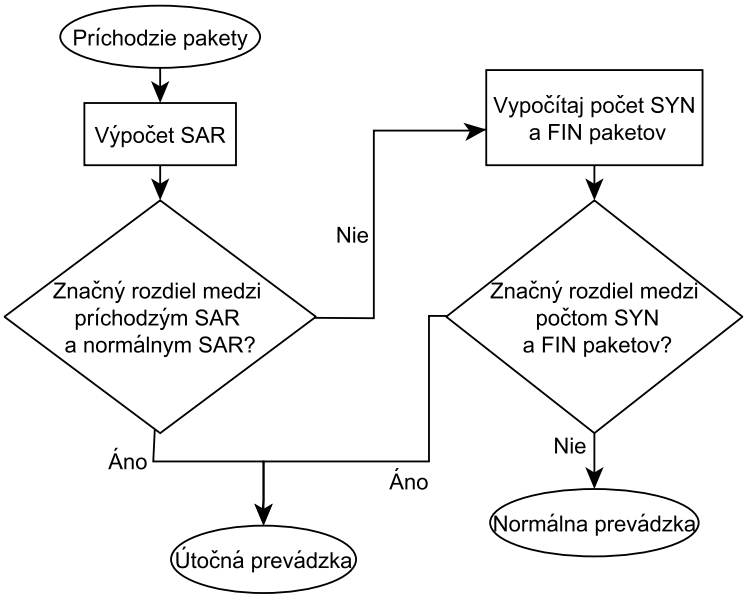
Dôvodom prečo sa vo vzorci používa deliteľ $(N-1)$ namiesto (N) je ten, že pri použití deliteľa N majú výpočty tendenciu produkovať neplnohodnotné výsledky rozptylu. Preto sa využíva ako deliteľ $(N-1)$, ktorý poskytuje korekciu tejto tendencie. Rovnicu 4.2 môžeme prepísať ako

$$\widehat{S}^2 = \frac{\sum_{i=1}^{N} X_i^2 - \frac{(\sum_{i=1}^{N} X_i)^2}{N}}{N-1}. \tag{4.3}$$

Výberová smerodajná odchýlka vzorky, $\widehat{S}$, je definovaná ako kladná odmocnina vzorky rozptylu $\widehat{S}^2$. Teda $\widehat{S} = \sqrt{\widehat{S}^2}$. Na zistenie či vzorka pochádza zo súboru normálnej prevádzky sa môže použiť Kolmogorov-Smirnov (K-S) test. Za testovacie hypotézy sa v tomto prípade môžu zvoliť:

H_0: Údaje o normálnom SAR (počet prijatých SYN – SYN arrival rates) majú predpokladané normálne rozdelenie.

H_1: Údaje o normálnom SAR (počet prijatých SYN – SYN arrival rates) nemajú predpokladané normálne rozdelenie[8].



Obr. 4.2: Diagram identifikovania prevádzky

4.3 Štatistická segregačná metóda

Základom pri návrhu štatistickej segregačnej metódy je určenie prahových hodnôt a správania medzi tokmi v prevádzke, k odlíšeniu legitímnej prevádzky od útočnej prevádzky, čo prispeje k riešeniu zabránenia veľkého množstva falošne–pozitívnych (prevádzka normálna vyhodnotená ako útočná) vyhodnotení. To stále zostáva výzvou v návrhu pre všetky metódy.

4.3.1 Vzorkovacia metóda

Veľa rôznych dostupných metód zameraných na útoky DDoS býva proaktívnych a reaktívnych [9]. Medzi metódami sa využíva jedna bežná metóda k detekcii TCP DDoS útoku a tou je kontrólny súčet prichádzajúcej prevádzky a odchádzajúcej prevádzky medzi ktorými je veľký rozdiel ako normálne. S pomocou tejto metódy dokáže aj dopravujúci smerovač detekovať DDoS útok. Pri ostatných DDoS útokoch sú dobrými indikátormi útoku napríklad náhly nárast v prevádzke alebo veľmi podobné správanie sa v prevádzke pre rôzne toky.

Ak sa potvrdí predbežná detekcia útoku tak prichádza na rad vzorkovacia metóda. Vzorkovacia metóda okamžite priradí samostatný čítač miery toku ku každej IP adrese v prevádzke. Čítač miery toku je určený tak aby zhromažďoval n vzoriek, kde vzorka je množinou všetkých prijatých paketov za sekundu. Pre účinnú detekciu, musí byť čas medzi zbieranými vzorkami presne stanovený. Avšak pozbieranie viacerých vzoriek vždy zvyšuje šancu na väčšiu presnosť, ale na druhú stranu sa v rovnakom časovom intervale spotrebúva viac času a výpočtového výkonu. K zníženiu intervalu detekcie tak aby odber vzoriek bol rýchly a efektívnejší, sa čítače uplatňujú len tri krát, každý v sekundovom intervale. Potom sú zozbierané len 3 vzorky pre IP adresu. Výsledkom je sada vzoriek zložená z ostatných vzoriek:

$$Sadavzoriek = \{vzorka1, vzorka2, vzorka3\}. \tag{4.4}$$

4.3.2 Analýza miery výskytu paketov

Všetky zozbierané vzorky sú následne priradené do segregačného štatistického mechanizmu. Tri vzorky sa potom medzi sebou porovnávajú, vzniká 3×3 možných kombinácii. Pred porovnávaním vzoriek musíme poznať u klienta skutočnú hodnotu normálnej miery výskytu paketov. Napríklad každý normálny klient môže generovať dve echo požiadavky pomocou ICMP (Internet Control Message Protocol) podobne ako každý skutočný klient má zabudovaný limit. Ak sa tento limit prekročí môže sa jednať o záplavu.

Zistenie útoku:

```
if |normal_vyskyt|<|vzorka1|<|vzorka2|<|vzorka3|
```

ak sa v prevádzke nachádza každá vzorka väčšia ako vzorka normálna tak sa potom v prevádzke môže nachádzať narastajúci DDoS útok,

```
if |normal_vyskyt|<|vzorka1|=|vzorka2|=|vzorka3|
```

ak sa v prevádzke nachádza vzorka, ktorá je väčšia ako normálna avšak rovnaká ako ostatné vzorky tak sa v prevádzke môže nachádzať konštantný DDoS útok.

Nepresvedčivé zistenia:
Všetky ostatné stavy sa zaraďujú do nepresvedčivých zistení, pretože tok môže byť legitímny alebo tiež môže niesť prvky DDoS útoku. Segregácie nie je tak ľahká, ako sa na prvý pohľad môže zdať a to hneď pre dva dôvody.

1. Pre stratu paketov v sieti útočná prevádzka nie je dostatočná k tomu aby naplnila požiadavky na zistenie útoku. Takejto prevádzke sa hovorí „neprimeraný útok“
2. Legitímna prevádzka sa môže javiť ako útok ak jej miera prijatých paketov je vyššia ako zvyčajne.

Použitie strednej aritmetickej hodnoty a smerodajnej odchýlky môže pomôcť pri segregovaní útoku od normálnej prevádzky. Stredná aritmetická hodnota je definovaná ako

$$\overline{X} = \frac{1}{3}\sum_{i=1}^{3} X_i, \tag{4.5}$$

kde X_i je počet paketov v i^{th} vzorky, $i = 1, 2, 3$.

$$\sigma = \sqrt{\frac{1}{3}\sum_{i=1}^{3}(X_i - \overline{X})^2}. \tag{4.6}$$

Presne povedané pre útok konštantnou mierov je smerodajná odchýlka okolo 0 a 1. Smerodajná odchýlka napomáha pri zistení útokov s malou mierov vysielaných paketov a útokov, ktoré sú klasifikované ako neprimerané, pomáha ich segregovať a potom pristupuje na rad korelačná analýza.

4.3.3 Korelačná analýza

Útoky vysielané za pomoci nízkej miery paketov od zombie počítačov majú medzi sebou podobnosť, ktoré je len zriedkavá u skutočného toku v normálnej prevádzke. Túto skutočnosť môžme objaviť pri korelácii toku. Z tohto dôvodu sa používajú kovariančné a korelačné analýzy aby sme zaistili segregáciu bežného toku od útočného pomocou analýzy podobností medzi tokom.

Kovariancia alebo spoločný rozptyl vyjadruje a opisuje závislosť medzi dvomi náhodnými veličinami.

Korelácia je väzba medzi dvomi alebo viacerými náhodnými veličinami. Preto sa korelácia používa medzi tokmi, pre tok x môže byť blízka alebo skoro rovnaká ako ďalšia premenná y. Vzorec je vyjadrený nasledovne

$$Cov(x, y) = \frac{\sum_{i=1}^{3}(X_i - \overline{X})(Y_i - \overline{Y})}{N}, \tag{4.7}$$

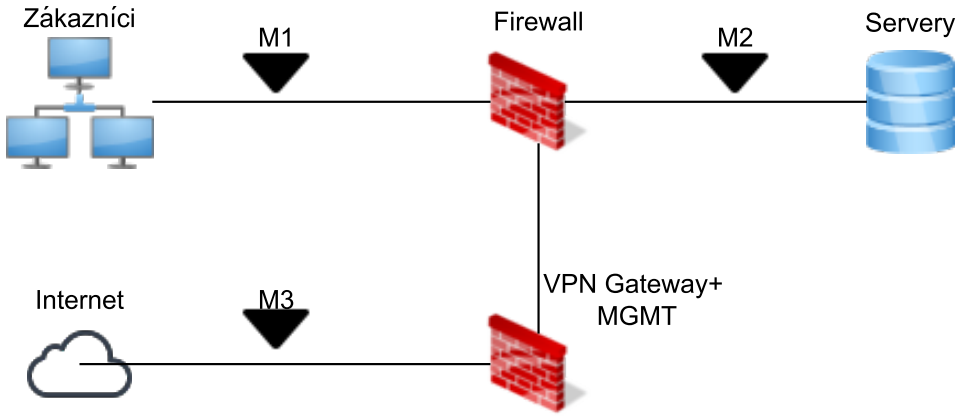
kde Y_i je počet paketov v i^th vzoriek ($i = 1, 2, 3$) pre tok y a $\overline{Y}$ je stredná aritmetická hodnota vzoriek pre tok y. Vzťah medzi koreláciou a kovarianciou je jednoducho znázornený na nasledujúcom vzorci

$$Correl(x, y) = \frac{Cov(x, y)}{\sigma_x \sigma_y}. \tag{4.8}$$

Korelácia pomáha k identifikácii podobných tokov a tak uľahčuje toky segregovať na neprimerané útoky a útoky s nízkou mierou vysielanie paketov[9][10].

5 ANALÝZA ZACHYTENÝCH ÚTOKOV SYN A UDP ZÁPLAVY

V tejto kapitole budú popísané zaznamenané útoky SYN a UDP flood, ktoré boli vygenerované pomocou nástroja *hping3* nainštalovanom na linuxovom OS (operačnom systéme – Operating System) **Kali**[1]. V kapitole budú popísané aj vlastnosti normálneho hodinového toku z 25.3.2015 19:00 zachyteného na sonde M2 v rámci školskej infraštruktúry Cesnetu. Na obrázku 5.1 môžete vidieť infraštruktúru uloženia sond.



Obr. 5.1: Umiestnenie záznamových sond v infraštruktúre Cesnetu

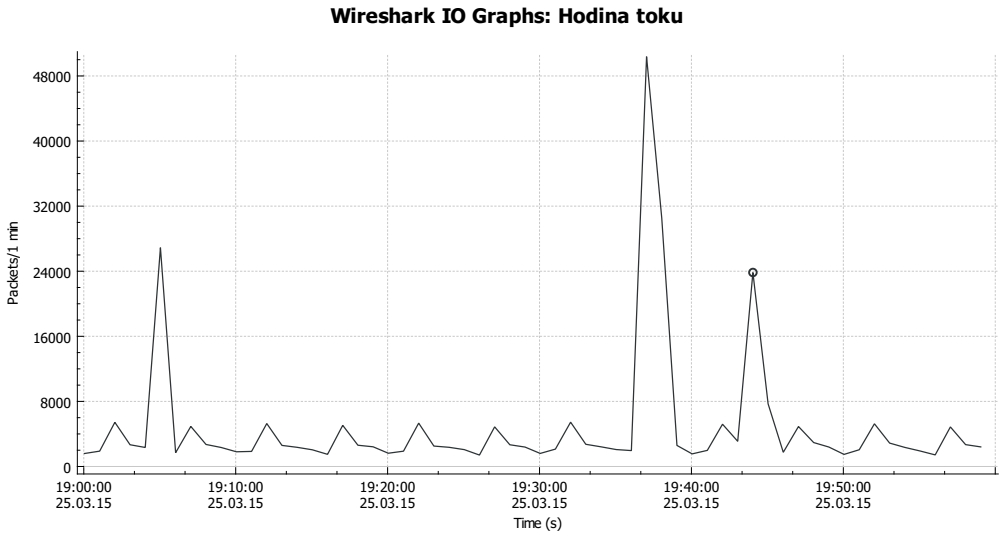
Záznam na sondách bol ukladaný v pravidelnom hodinovom intervale a dáta sú uložené na cloudovom úložišti Cesnetu.

5.1 Analýza hodinového toku

Záznam pochádzajúci z 25.3.2015 19:00 bol vybraná náhodne pre testovacie účely a je zachytený presne v časovom rozmedzí jednej celej hodiny. Záznam sa dá podľa potrieb upraviť aj na menšie celky a to napríklad programom *editcap*. Program je súčasťou aplikácie Wireshark[2], štandardne inštalovaný na linuxe Kali. Na sonde M2 bolo v čase od 19:00 do 20:00 hodiny zaznamenaná prevádzka zobrazená na obrázku 5.2.

Graf záznamu bol zhotovený pomocou programu Wireshark a tento tok bude aj analyzovaný pomocou tohto programu. V grafe môžeme vyčítať, že v hodine

[1]Linuxový OS na stiahnutie zdarma na stránke https://www.kali.org/downloads/
[2]Aplikácie je zdarma na stiahnutie na stránke https://www.wireshark.org/



Obr. 5.2: Graf celkového počtu paketov prevádzky od 19:00 do 20:00 hodiny

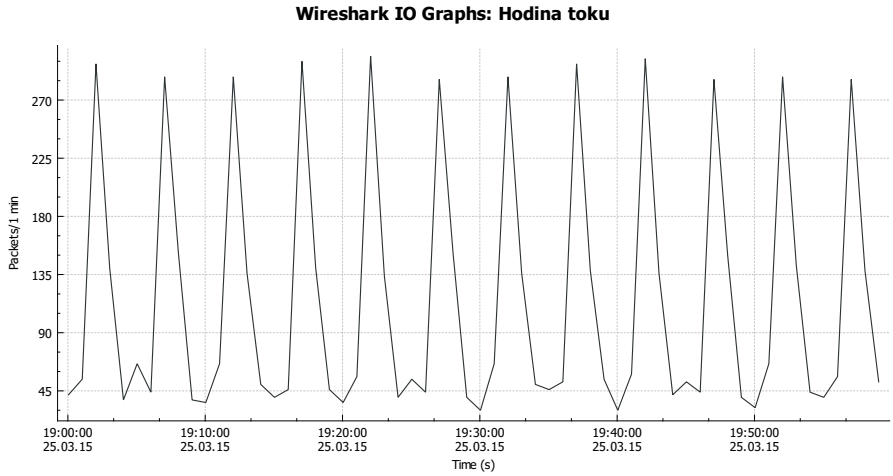
tohto toku je špička 42000 paketov za minútu. Pre rozdelenie toku môžme použiť spomínaný príkaz a rozdeliť tok napríklad na minútové (60 sekundové) intervaly. Syntax príkazu by vyzeral nasledovne:

```
editcap -i 60 hodinatoku.pcap hodinatokuvystup.pcap
```

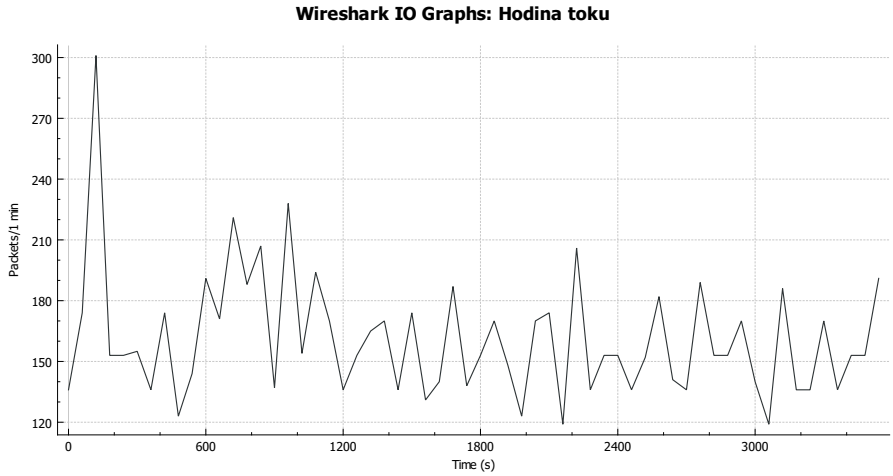
Rozdelenie toku je výhodné pre jeho detailnejšie skúmanie. Celkovo prenesených paketov v tejto hodine toku bolo 291865 a priemerne za minútu 523,31 z toho pakety prenesené vo veľkostných intervaloch:

- 40–79 kB – 147962 priemerne za minútu 64,88,
- 80–159 kB – 40116 priemerne za minútu 106,88,
- 160–319 kB – 4382 priemerne za minútu 221,95,
- 320–639 kB – 830 priemerne za minútu 460,94,
- 640–1279 kB – 10188 priemerne za minútu 1082,96,
- 1280–2559 kB – 88171 priemerne za minútu 1432,99.

V hodinovom toku bolo zaznamenaných 6896 TCP paketov s príznakom SYN, ktorých minútový interval je zobrazený v grafe 5.3, 154360 paketov s príznakom ACK a 5188 paketov s príznakom ACK+SYN. V tomto toku sa vyskytlo 9677 UDP paketov zobrazených v minútovom intervale na grafe 5.4 a ďalších, ktoré nie sú predmetom skúmania v práci.



Obr. 5.3: Graf počtu TCP paketov s príznakom SYN od 19:00 do 20:00 hodiny



Obr. 5.4: Graf počtu UDP paketov od 19:00 do 20:00 hodiny

Testovanie útokov SYN záplavy a UDP záplavy prebiehalo za pomoci dvoch virtuálnych počítačov s operačným systémom Windows 7 profesional[3] a Kali Linux.

Na virtuálnom počítači s operačným systémom Windows 7 profesionál (obeti útoku) bol nainštalovaný program Wireshark na zachytávanie prevádzky a program *Ostinato*[4] vďaka ktorému bola generovaná virtuálna prevádzka na sieti.

Útočník virtuálny počítač s operačným systémom Kali Linux produkoval pomocou programu hping3 útočnú prevádzku voči obeti útoku druhému virtuálnemu počítaču.

[3]Originálna verzia Windowsu bola zadarmo stiahnutá pre študentov VUT vďaka Microsoft DreamSpark

[4]Voľne dostupný na stiahnutie na adrese http://ostinato.org/

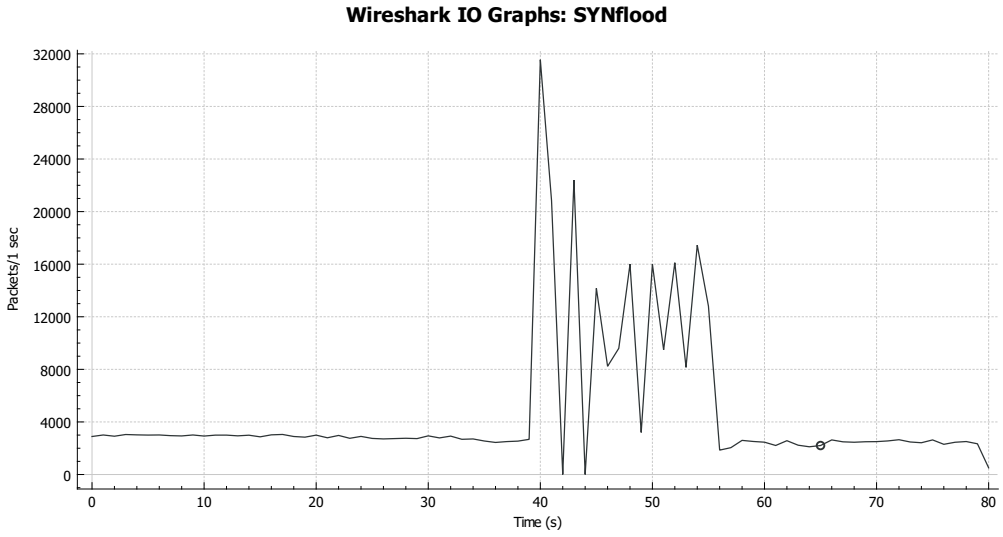
Syntax príkazu pre vygenerovanie útok TCP SYN flood je nasledovný[11]

```
hping3 --rand-source 192.168.101.131 --flood -S -L 0 -p 80
```

kde

- IP adresa `192.168.101.131` je adresou obete útoku,
- parameter `-rand-source` určuje, že pakety budú vysielané z rôznych zdrojových adries,
- parameter `-flood` posiela pakety najvyššou možnou rýchlosťou,
- parameter `-S` určuje, že TCP pakety budú mať príznak SYN,
- parameter `-L` nastavuje TCP ACK,
- parameter `-p` slúži k nastaveniu cieľového portu, primárne je nastavený na 0.

Na počítači obeti bol zapnutý program Wireshark tak isto ako aj program Ostinato. Pomocou programu Ostinato bola generovaná prevádzka 3000 TCP paketov s príznakom SYN+ACK za sekundu pre lepšiu priehľadnosť grafu. Po zapnutí zadaní príkazu hping3 na útočníkovom počítači začal príkaz generovať SYN pakety na port cieľa 80. Na počítači obeti bolo citeľné spomalenie reakcie systému, tak isto aj odpoveď na požiadavky bola spomalená. Tento test bol vyskúšaný aj na cvičnom domácom smerovači kde po chvíli došlo k vypadnutiu služby a nemožnosti sa pripojiť na toto zariadenie. SYN flood útok je bližšie popísaný v sekcii 2.2.2



Obr. 5.5: Graf testovacieho útoku SYN záplavy

Útok bol generovaný 14 sekúnd a stropu dosiahol hneď po spustení kde vygeneroval 32000 paketov za sekundu čo môžme vidieť v grafe 5.5.

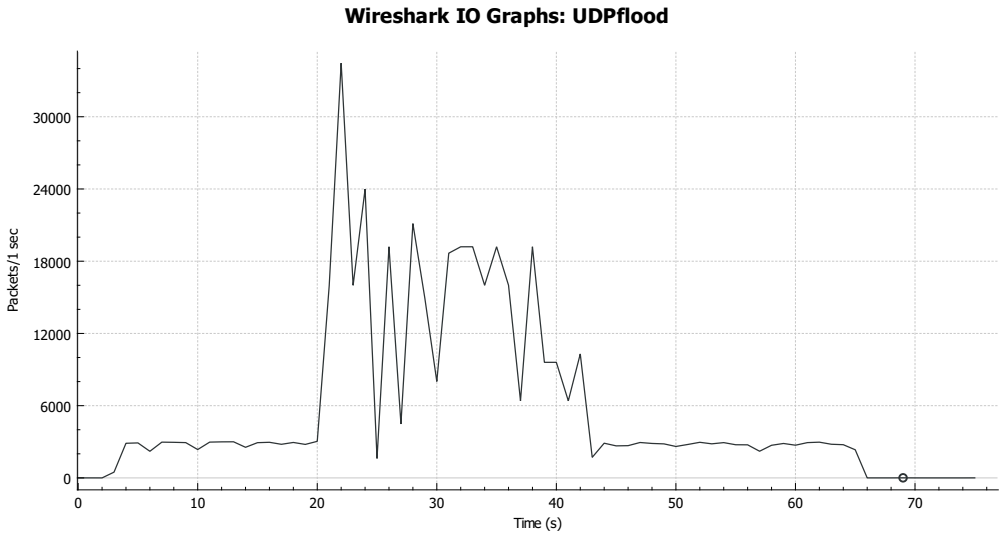
Na testovanie UPD záplavy bol taktiež použitý spomínaný program *hping3*. Tento krát boli generované UDP pakety na port 53 určený pre službu DNS (systém doménových mien – Domain Name Service). Syntax príkazu bol nasledovný [11]

```
hping3 -udp -p 53 -flood -rand-source 192.168.101.131
```

kde

- parameter `-udp` určuje druh útoku v tomto prípade útoku UDP,
- parameter `-p` slúži k nastaveniu cieľového portu, primárne je nastavený na 0,
- parameter `-flood` posiela pakety najvyššou možnou rýchlosťou,
- parameter `-rand-source` určuje, že pakety budú vysielané z rôznych zdrojových adries,
- IP adresa `192.168.101.131` je adresou obete útoku.

Tak isto ako aj v prípade SYN záplavy bol na počítači obete zapnutý program Wireshark a program Ostinato. V programe Ostinato bola tento krát zvolená generovaná prevádzka s UDP paketmi a tak isto ako v prípade SYN záplavy množstvo generovaných paketov bolo 3000 paketov za sekundu. Útok má podobný priebeh ako pri útoku SYN záplavy avšak pri útoku sa využívajú odlišné parametre popísané v 2.1.1.



Obr. 5.6: Graf testovacieho útoku UDP záplavy

Útok bol generovaný po dobu 21 sekúnd a najvyšší počet paketov ktorý stanica obete prijala bol 35000 v jednej sekunde znázornené na grafe 5.6.

V porovnaní s normálnou prevádzkou zo sondy M2 sú tieto hodnoty prijatých paketov v jednej sekunde alarmujúci vysoké a teda jasne indikujúce útoky DoS.

6 NÁVRH A POPIS PROGRAMU

Program na detekciu záplavových DDoS útokov bol navrhovaný na prácu so statickými metódami, kde sa program samostatne naučí o správaní sa siete a sieťového toku. Program je programovaný v jazyku C++, pracujúci pod linuxovými distribúciami. Cieľom programu je zachytiť záplavové útoky DDoS. K programu je pridaný aj súbor **README.md** v ktorom sú popísané balíčky, potrebné na nainštalovanie a metódy spustenia programu. Program má dve časti

1. *Učiaca časť* – v learning alebo učiacej časti sa program učí o správaní sa na sieti, detekuje príchodzie pakety a zapisuje ich do štatistiky, ktorú ďalej využíva k porovnávaniu v detekčnej časti
2. *Detekčnú časť* – v detekčnej časti program pracuje s dopredu naučenou štatistikou o správaní sa toku v sieti a pri prekročení naučených štatistík detekuje útok.

Program sa skladá zo šiestich spustiteľných parametrov, ktoré je možné medzi sebou kombinovať. Základný príkaz na spustenie programu je

```
./build/xmisko03
```

Pre spustenie programu je potrebné byť v zložke kde sa nachádza zložka build a samotné časti programu. Možné parametre, ktoré slúžia na určenie programu čo od neho práve vyžadujeme sú

- parameter `-l` určuje programu, že sa jedná o mód učenia,
- parameter `-c` slúži programu k určeniu konfiguračného súboru, ktorý sa teraz vytvorí ak je zadaný s parametrom *-l* a v prípade detekcie určuje súbor vďaka ktorému sa bude porovnávať odchytávaná prevádzka s naučenou štatistikou
- parameter `-f` slúži k tomu aby program rozpoznal, že sa jedná o učenie sa zo súboru
- parameter `-i` nastavuje interface, na ktorom budeme prevádzku sledovať alebo odchytávať
- parameter `-h` show help, zobrazí nám všetky parametre a popis k ním
- parameter `-t` čas pre ukončenie módu učenia sa v sekundách

6.1 Učiaca časť

Časť učenia pozostáva z dvoch hlavných zložiek, ktorými sú

- učenie sa zo súboru pcap
- učenie sa z vybraného interface portu

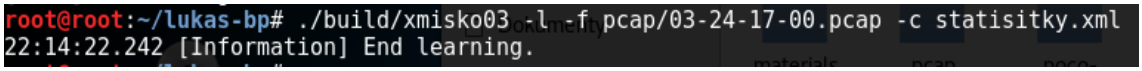
Pri **učení sa zo súboru** nahráme pomocou programu dopredu odchytený súbor pcap napríklad pomocou programu *Wireshark* alebo pomocou paketového analyzátoru v linuxovej distribúcii *tcpdump*. Súbor pcap uložený v PC si nahráme pomocou príkazu

```
./build/xmisko03 -l -f pcap/03-24-17-00.pcap -c statistika.xml
```

kde

- parameter `-l` určuje programu, že sa jedná o mód učenia,
- parameter `-f` slúži k tomu aby program rozpoznal, že sa jedná o učenie sa zo súboru
- parameter `-c` slúži programu k určeniu konfiguračného súboru, ktorý sa teraz vytvorí.

Po úspešnom vytvorení štatistiky nám program vypíše v terminály oznámenie zobrazené na obrázku 6.1.



Obr. 6.1: Výpis z terminálu pri úspešnom vytvorení štatistiky

Pri **učení sa z portu** definuje port pomocou parametra *-i* na ktorom budeme chcieť prevádzku zachytávať a zaznamenávať.

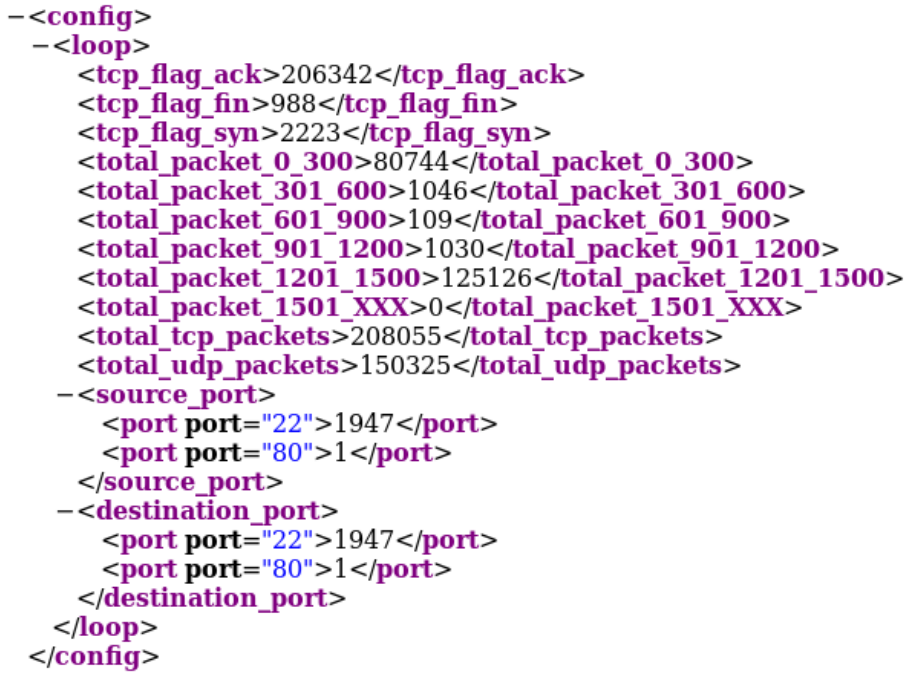
Príklad použitia programu pri učení sa z portu *eth0*

```
./build/xmisko03 -l -i eth0 -c statistika.xml -t 60
```

kde

- parameter `-l` určuje programu, že sa jedná o mód učenia,
- parameter `-i` slúži k tomu aby program rozpoznal, že sa jedná o učenie sa z portu
- parameter `-c` slúži programu k určeniu konfiguračného súboru, ktorý sa teraz vytvorí.
- parameter `-t` čas pre ukončenie učenia sa v sekundách

Vytvorený súbor *statistika.xml* je vo formáte xml pre lepšiu čitateľnosť. Príklad vytvoreného súboru *statistika.xml* z pcap súboru *http.cap* je zobrazený na obrázku 6.2



Obr. 6.2: Vytvorený súbor štatistík z pcap v cykle 30 sekúnd

Takto vytvorené štatistiky si užívateľ môže podľa potreby upraviť a zväčšiť poprípade zmenšiť objem detekovaných dát, v prípade, že počas štatistiky na serveri práve prebiehal útok alebo anomália.

Vytvorenie štatistiky funguje na princípe čítania paketov v určitý časový interval, ktorý je možné si definovať v časti programu *Statistics.cpp*.

V statistics.cpp je možné nastaviť 3 hodnoty v intervale, ktorými sú:

- `const int measurementCycle`
  – v tejto časti sa nastavuje hodnota v sekundách po akej sa má cyklus čítania paketov znova vykonať
- `const int measurementCycleMin`
  – začiatok merania hodnota v sekundách
- `const int measurementCycleMax`
  – koniec merania hodnota v sekundách

Pri nastavení hodnôt:

- `const int measurementCycle = 20`
- `const int measurementCycleMin = 0`
- `const int measurementCycleMax = 15`

Cyklus bude zaznamenávanie hodnôt vykonávať každých *dvadsať sekúnd*. Začiatok čítania paketov začne v *nultej sekunde* a skončí v *pätnástej sekunde*. Program *päť sekúnd* počká a celý proces sa zopakuje znova. Na rovnakom princípe funguje aj detekcia zachytávaných paketov popísaná v sekcii 6.2

Zo zaznamenávaných paketov sa potom spravia štatistiky pomocou jednoduchého priemeru. Všetky pakety zachytené v jednom cykle sa zaznamenávajú do premenných akými sú: *tcp_ack_flag, tcp_syn_flag, tcp_fin_flag* a podobne.

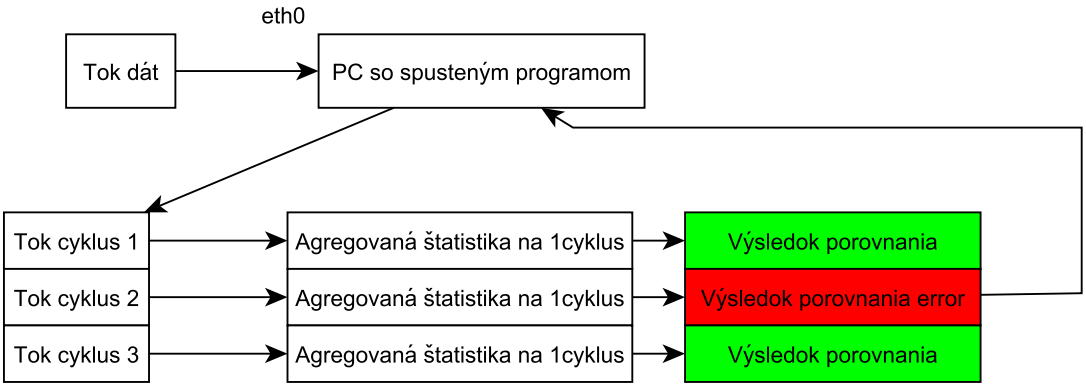
Štatistiky sú tvorené systémom priemeru. V prvom cykle sa pakety zaznamenajú do premennej c1.*názov_premennej*. V druhom cykle sa pakety zaznamenajú do premennej c2.*názov_premennej*. Nakoniec sa premenné c1.*názov_premennej* a c2.*názov_premennej* následne sčítajú a spraví sa ich priemer podľa vzorca 6.1.

$$c1.názov\_premennej = \frac{c1.názov\_premennej + c2.názov\_premennej}{2}. \tag{6.1}$$

Tento cyklus pokračuje až do zastavenia učiaceho sa módu, buď užívateľom pomocou *ctrl+c* alebo pomocou parametra *-t* pri metóde učenia sa z portu a pri učení sa zo súboru po prejdení celého súboru pcap.

6.2 Detekčná časť

V detekčnej časti programu sa program podľa zadaných kritérií rozhoduje či v sebe tok dát nesie útočnú prevádzku alebo nie. Detekčná časť funguje na rovnakom princípe rozdelenia toku do cyklov ako bolo popísané v učiacej časti 6.1.



Obr. 6.3: Detekovanie toku dát na porte eth0

Ako môžme sledovať na obrázku 6.3 tok dát prichádza na port eth0 nášho počítača na ktorom je spustený program na detekciu útokov.

Program tok dát rozdeľuje na cykly podľa nastavených parametrov, ktorými sú:

- `const int measurementCycle`
- `const int measurementCycleMin`
- `const int measurementCycleMax`

Tieto parametre sú zhodné s parametrami nastavenými pre vytvorenie štatistiky v učiacom sa móde. Program nám rozdelí tok na viacero cyklom v určitých časových intervaloch, ktoré si môže užívateľ určiť v súbore programu *Statistics.cpp*.

Program toky dát rozdelené do cyklov uchováva v RAM pamäti až do kým sa cyklus neporovná s agregovanou štatistikou pre cyklus a nenahradí druhým nasledujúcim cyklom, ktorý sa opäť porovnáva s agregovanou štatistikou pre cyklus. Tento proces sa vykonáva neustále kým užívateľ neukončí program klávesami *ctrl+c*.

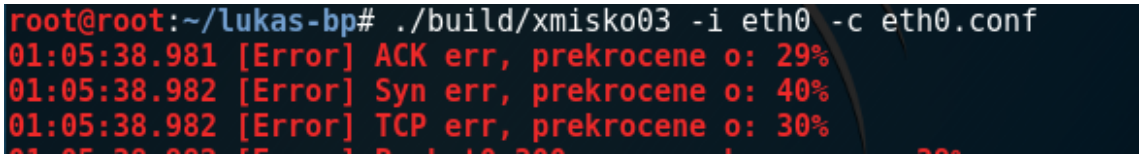
Ak program zaznamená zvýšenie sa jednej z prednastavených veličín toku oproti štatistikám vypíše v terminály chybu a zaznamená ju do logu, ktorý sa ukladá do zložky s programom. Chyby sú rozdelené na dva druhy:

1. Chyba error – chyba error nastane v prípade ak daná komunikácia prekročí nad 20% a zároveň nedosiahne hodnoty 50% vytvorenej štatistickej hodnoty
2. Chyba critical – chyba critical nastane v prípade ak daná komunikácia prekročí nad 50% vytvorené štatistické hodnoty

Tieto definované percentuálne chyby sú len pre testovacie účely. Administrátor siete alebo užívateľ programu, ktorý vie aké veľké zaťaženie si server môže dovoliť, môže podľa vlastného uváženia nastaviť percentuálny výpis chýb.

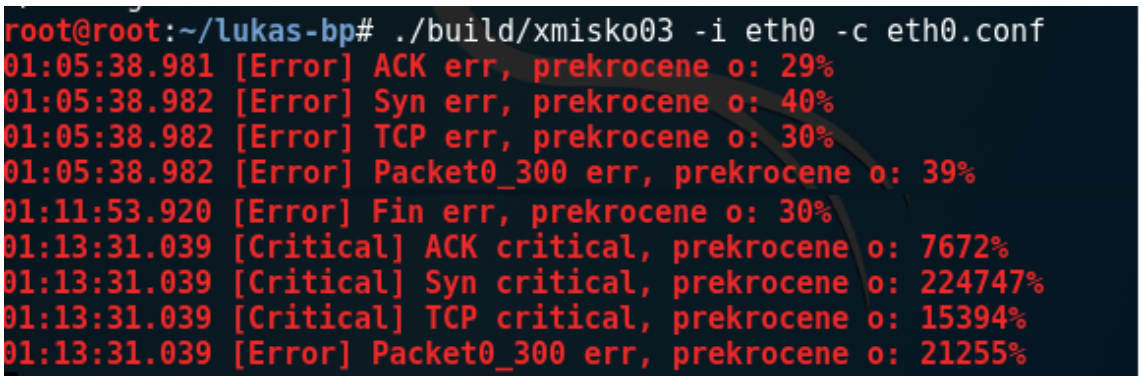
Program pracuje v terminály a porovnáva prichádzajúce pakety s odchytenou prevádzkou, ktorá je zapísaná vo vytvorenom súbore štatistík, ktorý sa vytvára pomocou príkazov v programe. Pri detekcii prekročenia štatistických limitov vypíše program na okno terminálu chybu podľa veľkosti prekročenia limitu.

Pri chybe ktoré je v rozmedzí **20%** až **50%** program vypíše chybu error znázornenú na obrázku 6.4.



Obr. 6.4: Detekcia 20% vzrastu ACK,FIN A SYN na porte eth0

Pri chybe ktorá prekročí hranicu **50%** program vypíše chybu critical znázornenú na obrázku 6.5.

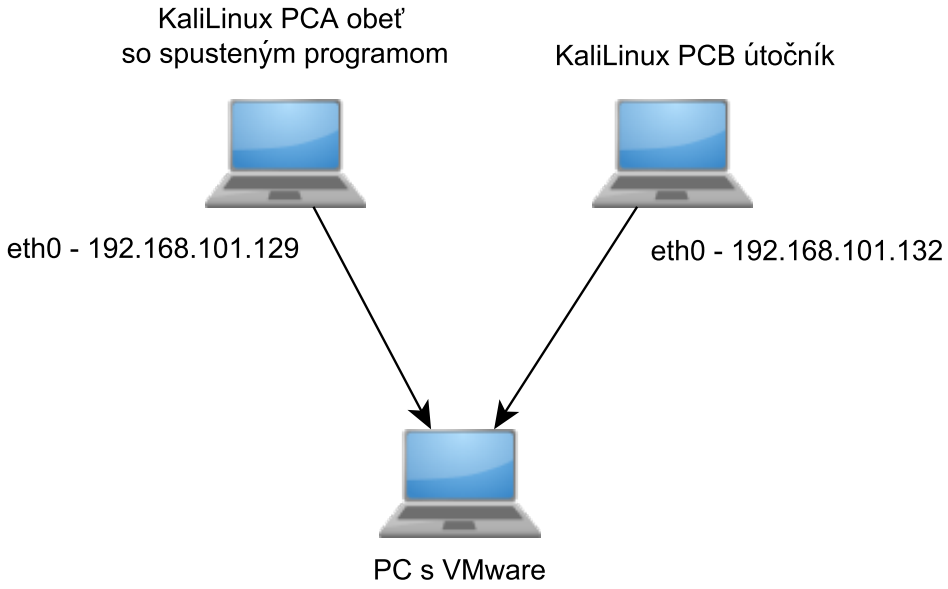


Obr. 6.5: Detekcia vzrastu hodnôt o viac ako 50% na porte eth0

Všetky vypísané chyby zobrazené na terminály sa rovnako exportujú do logu, ktorý je v priečinku kde je uložený program pod názvom ***logger.log***. Chyby zapisované do logu, môžu dosiahnuť hodnotu až 1MB a po prekročení tejto hodnoty sa log prepisuje znova od začiatku. Možnosti nastavenia veľkosti logu a jeho uloženia nájdeme v konfiguračnom súbore *logging.ini*.

7 TESTOVANIE PROGRAMU V LINUXOVEJ DISTRIBÚCII KALI

Testovanie pribiehalo na jednom počítači, medzi dvoma virtuálnymi strojmi zobrazených na obrázku7.1na ktorých bol spustený Linux distribúcie Kali vo virtuálnom prostredí *VMware Workstation*[1]. Program je schopný detekovať prevádzku po prekročení 20% hodnôt zapísaných v súbore štatistík, okamžite po uplynutí nastaveného cyklu a porovnaní hodnôt zachytených s hodnotami zaznamenanými v štatistikách.



Obr. 7.1: Zapojenie virtualnych strojov na PC pomocou VMware

Pri testovaní bola odchytávaná prevádzka z PCA na ktorú bola použitá metóda učenia sa z kapitoly 6.1 následne z ktorej bola vypracovaná štatistika *eth0.conf*. Štatistika bola odchytávaná v tridsať sekundovom intervale, po dobu pätnástich minút. Počas doby pätnástich minút bol počítač PCA ovládaný užívateľom, boli otvárané webové stránky, spúšťané videá aby sme zaistili zaznamenanie štatistík normálneho používania.

```
./build/xmisko03 -i eth0 -c eth0.conf -l -t 900
```

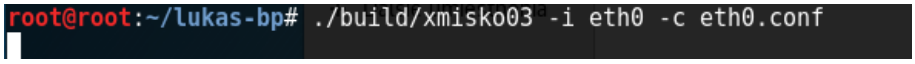
Po odchytení a zaznamenaní štatistiky do súboru *eth0.conf* prebehlo spustenie programu v detekčnom móde na porte eth0. Jedná sa o rovnaký typ štatistiky ako

[1]Program VMware Workstation bol stiahnutý zo stránky https://www.vmware.com/

štatistika s koncovkou .xml. Program vždy vygeneruje štatistiku v podobe xml súboru. Názov bol zvolený pre testovacie účely, vzhľadom k tomu aby bolo jasné, že sa jedná o konfiguračný súbor z ktorého budeme porovnávať prevádzku.

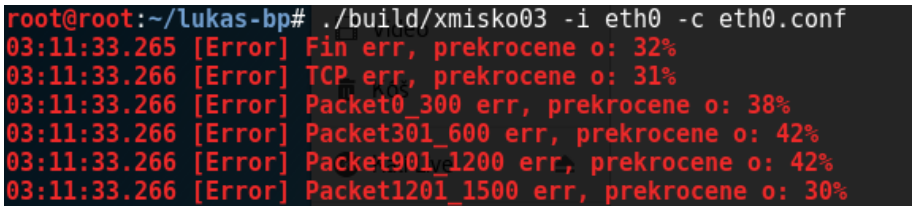
```
./build/xmisko03 -i eth0 -c eth0.conf
```

Program nevypisuje žiadne známky prebiehajúcej anomálie ako môžme vidieť na obrázku7.2

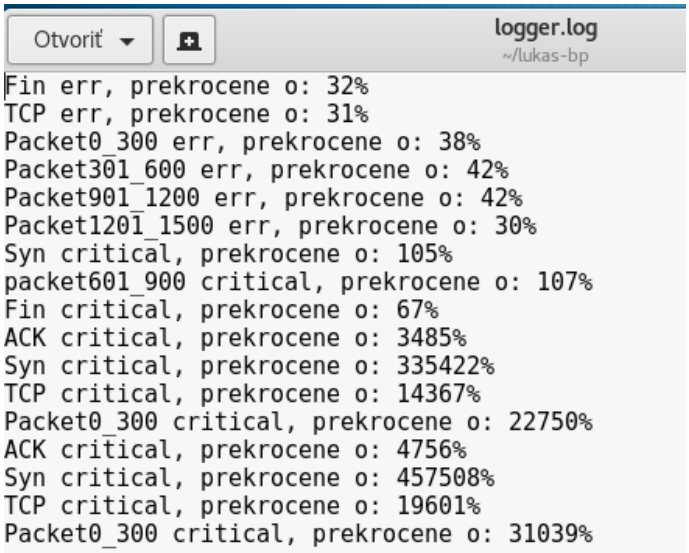


Obr. 7.2: Spustený program bez známok anomálii

Pri zaznamenaní anomálii menšej ako 50%, čiže pre testy definovanej v rozmedzí 20% až 50% program vypíše error do okna terminálu s percentuálnou odchýlkou zobrazenom na obrázku7.3, ktorá sa zaznamená do logu *„logger.log“* zobrazenom na obrázku7.4.



Obr. 7.3: Zaznamenanie anomálie v rozmedzí 20% až 50%

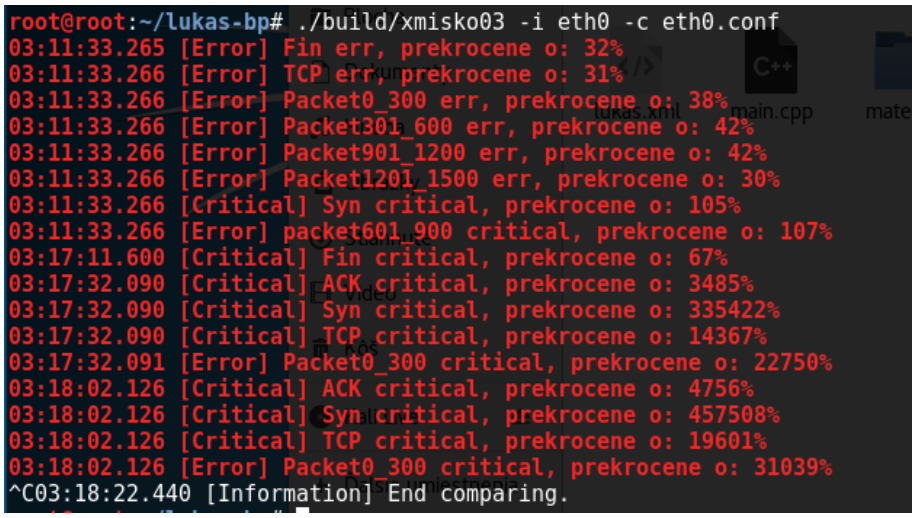


Obr. 7.4: Hodnoty zapísané z terminálu do logu

Z PCB, čiže z útočného PC pošleme SYN flood útok na hostiteľský počítač na ktorom je spustený program na zachytávanie útokov v sieti pomocou príkazu

```
hping3 --rand-source 192.168.101.129 --flood -S -L 0 -p 80
```

Program v pravidelných cykloch, pre testovacie účely, definovaných na tridsať sekúnd odchytáva komunikáciu a porovnáva ju so štatistikou vytvorenou rovnako pre tridsať sekúnd. Program následne vyhodnotí aké limity boli prekročené a vypíše ich na okno terminálu a rovnako aj zaznamená do logu zobrazenom na obrázku7.4. Vďaka tomuto výpisu môžme vidieť, že program zaznamenal útok a pomer paketov sa zvýšil o 3485% pre TCP ACK, oproti našej normálnej štatistike, pri bežnej práci s PCA.



Obr. 7.5: Zaznamenanie útoku pred a po prekročení 50%

Pre zrušenie detekčného módu je používaná skratka *ctrl+c*, kde po stlačení skratky vidíme výpis „End comparing“ zobrazený v poslednom riadku na obrázku7.5

8 ZÁVĚR

Cieľom tejto bakalárskej práce bolo preskúmanie využitia statických metód a časových radov k identifikácii DDoS útokov a následné vytvorenie softwaru, ktorý je schopný útoky detekovať.

V teoretickej časti boli preskúmané najčastejšie druhy útokov hlavne útokov odmietnutia služby DoS, DDoS, DRDoS, ich správanie sa, zámery týchto útokov a nebezpečenstvo s nimi spojené. Ďalej sa v práci popisuje kategorizácia týchto útokov podľa miery výskytu paketov za určitý časový interval. Tu sú tieto útoky graficky znázornené a popísané.

V práci sú detailne popisované protokoly UDP a TCP a ich možné zneužitie k záplavovým útokom typu UDP flood a SYN flood, ktoré sú spojené s útokmi na odmietnutie služby. Ďalej sa v práci nachádza rozbor troch statických metód, ktoré využívajú štatistické analýzy.

V ďalšej časti práce sú analyzované a graficky spracované údaje z jednej hodiny merania sondy M2 v školskej infraštruktúre Cesnetu. Ďalej sa tu popisujú použité nástroje na analýzu a nástroje využité k cvičnému testovaniu generovaného útoku medzi dvoma virtuálnymi počítačmi s operačnými systémami Windows 7 profesinál a Kali Linux. Analýzou a porovnaním záznamov sme si potvrdili obrovské anomálie v sieťovom toku bez útoku a počas útokov.

V záverečnej časti práce je popísaný vytvorený software pracujúci na linuxových distribúciach, ktorý je schopný na základe naučených štatistík detekovať príchodzie útoky a tie vypísať na terminál a do logu. Program je ľahko nastaviteľný a prispôsobiteľný rôznym požiadavkám siete alebo užívateľa. V programe sú zhotovené jednoduché štatistiky, ktoré v ďalšom pokračovaní práce na programe možno rozšíriť aby bol program čo najpresnejší.

Funkčnosť programu je značná avšak pre dokázanie jednoznačnej schopnosti detekcie útočného toku sú nutné ďalšie testovania s väčším zaťažením siete a programu. Ako môžeme vidieť aj jednoduchá štatistika v statických metódach je využiteľná pri základnej detekcie útočného toku v sieti. Nie vždy sa v prevádzke musí jednať o útočný tok, niekedy je to len anomália v sieti. Pre reálne nasadenie programu do skutočnej siete a prevádzky je nutné zhotovenie presnejších štatistických metód na základe pozorného skúmania siete a správania sa sieťového toku. Pre ďalšiu prácu by bolo vhodné tieto štatistické metódy upraviť, pridať do programu viaceré premenné vďaka ktorým je program schopný detekovať útoky a tak zlepšiť schopnosť programu detekovať a rozlišovať skutočné útoky od anomálii na sieti a v sieťovom toku. Program je uložený na priloženom CD.

LITERATÚRA

[1] Tech Target. *Tech Target Network Security* [online]. 2007, posledná aktualizácia 11. 10. 2007 [cit. 19. 11. 2016].
Dostupné z URL: <http://searchsoftwarequality.techtarget.com/definition/denial-of-service>.

[2] PENG, Tao, Christopher LECKIE a Kotagiri RAMAMOHANARAO. *Detecting Distributed Denial of Service Attacks Using Source IP Address Monitoring* [online]. In: Proceedings of the Third International IFIP-TC6 Networking Conference (Networking (2004), 2002, 1–14 [cit. 19. 11. 2016]. DOI: 10.1.1.13.5997. Dostupné z URL: <http://citeseerx.ist.psu.edu/viewdoc/summary?doi=10.1.1.13.5997>.

[3] HALLER, Martin. *Denial of Service (DoS) útoky: záplavové typy* [online]. In: Lupa, 2006 [cit. 19. 11. 2016]. Dostupné z URL: <http://www.lupa.cz/clanky/denial-of-service-dos-utoky-zaplavove-typy/>.

[4] ENDORF, Carl F., Eugene SCHULTZ a Jim MELLANDER. *Detekce a prevence počítačového útoku* [kniha]. Praha: Grada, 2005, 355s. [cit. 19. 11. 2016]. ISBN 80-247-1035-8.

[5] UDP Flood. *Incapsula* [online]. [cit. 20. 11. 2016]. Dostupné z URL: <https://www.incapsula.com/ddos/attack-glossary/udp-flood.html/>.

[6] MATETI, Prabhaker. *TCP Exploits* [online]. [cit. 03. 12. 2016]. Dostupné z URL: <http://cecs.wright.edu/~pmateti/Courses/4420/TCPexploits/index.html/>.

[7] HANÁK, Jiří. *Postavte se velké vodě: Nejčastějšími DDoS útoky jsou SYN a UDP záplavy* In: Master [online]. 2015 [cit. 04. 12. 2016]. Dostupné z URL: <https://www.master.cz/blog/jak-funguji-ddos-utoky-typy-ddos-syn-udp-zaplavy//>.

[8] CHEN, Chin-Ling. A New Detection Method for Distributed Denial-of-Service Attack Traffic based on Statistical Test. *Journal of Universal Computer Science.* 2009, 2009(2), 488–504. DOI: 10.3217/jucs-015-02-0488. [cit. 10. 12. 2016]

[9] UDHAYAN, J. a T. HAMSAPRIYA. Statistical Segregation Method to Minimize the False Detections During DDoS Attacks. *International Journal of Network Security.* 2011, 2011(3), 152—160. [cit. 10. 12. 2016]

[10] SIEGRIST, Kyle. 5. Covariance and Correlation. In: *Math.uah.edu*[online]. ©1997–2015 [cit. 11. 12. 2016]. Dostupné z URL: <`http://www.math.uah.edu/stat/expect/Covariance.html//`>.

[11] SALVATORE, Sanfilippo. Hpin3(8) – Linux man page. In: *Die*[online]. [cit. 11. 12. 2016]. Dostupné z URL: <`https://linux.die.net/man/8/hping3//`>.

ZOZNAM SYMBOLOV, VELIČÍN A SKRATIEK

DoS odmietnutie služby – Denial of Service

DDoS distribuované odmietnutie služby – Distributed Denial of Service

DRDoS distribuovano reflektované odmietnutie služby – Distributed Reflector Denial of Service

UDP užívateľský datagramový protokol – User Datagram Protocol

TCP protokol riadenia prenosu – Transmission Control Protocol

ISN Initial Sequence Number

CPU centrálna procesorová jednotka – Central Processing Unit

SIM monitorovanie zdrojovej IP adresy – Source IP address Monitoring

IAD databáza IP adries – IP Address Database

SAR počet prijatých SYN – SYN arrival rates

ICMP Internet Control Message Protocol

OS operačnom systéme – Operating System

DNS systém doménových mien – Domain Name Service